# Task 3

## A place to save the data

The first thing I will do is choose a place to store the data.  
Because the data about the features and the user names is not classified I will create an excel csv file that will give us a faster and more compatible way to interact with the data especially in powershell.

## Users

As for the users I will save their secrets in a secure place such us aws secret manager or an encrypted file saved in a s3 bucket.

## Features

As shown in the previous task saving our list in a designated csv file will help us not only list our desired features but also help us track down the features that installed correctly and the one that didn’t.

**All of those csv files will be save in a remote git hub repository as part of our code and will be pulled as well.**

## The solution itself

### The feature code

As for the feature code it is quite simple and very straight forward I will use the following commands:

Import-csv -path "examples\complete\Powershell-codes\Features and users.csv"  
$Features = $data.Features | Where-Object {$\_.trim() -ne "" }

And from here we will use our code created in the second question.

Install-Feature -Features $Features -ComputerName $computername -Username $username -password $password

### The users code

As for the User code we will create a function with a proper documentation about its parameters and overall usage.  
In our case the key command that will be run is   
New-LocalUser -Name $user

With the username given in the csv list.

### Users Secrets

As for the user secrets as mentioned above we will use the aws secret manager to store the users secrets such as password and other sensitive information and we will pull it with powershell modules to be used in our scripts.

## Conclusion

My final goal is to make the entire process of creating users and features as automated as possible in consideration of security and efficiency.